

Claims:

What is claimed is:

1. A network having a plurality of nodes, comprising:
an event channel adapted an event between a publisher node and a subscriber node within said network;
a filter to identify said event on said subscriber node; and
5 an application on said subscriber node to receive said event according to said filter.
2. The network of claim 1, further comprising an event server on said subscriber node, said event server adapted said event to said filter from said event channel.
3. The network of claim 2, wherein said event server exchanges information with another event server.
4. The network of claim 2, wherein said application opens said event channel through said event server.
5. The network of claim 1, wherein said event includes at least one pattern field, wherein said pattern field matches a filter field within said filter.

6. The network of claim 5, wherein said event further includes a data field.
7. The network of claim 1, wherein said event channel has a unique name.
8. The network of claim 7, wherein said unique name is registered in a naming service within said network.
9. The network of claim 2, wherein said publisher node has a configuration, said configuration being known to said event server on said subscriber node.
10. The network of claim 1, further comprising an event server on said publisher node, wherein said event server publishes said event on said event channel.
11. The network of claim 10, wherein said subscriber node has a configuration, said configuration being known to said event server on said publisher node.
12. A node within a network to exchange information, comprising:
 - an application;
 - an event server adapted to receive an event from an event channel; and
 - a filter to identify said event for said application.

13. The node of claim 12, wherein said event server includes an event control block to subscribe to said event channel.
14. The node of claim 13, wherein said event server further includes an event control manager to control said event control block.
15. The node of claim 14, wherein said event control manager updates said event control block.
16. The node of claim 14, wherein said event control manager detects an overload condition within said event control block.
17. The node of claim 14, wherein said event control manager controls a configuration of said event control block.
18. The node of claim 13, wherein said event server further includes an event protocol module to manage network connections to said event control block.
19. The node of claim 13, wherein said event control block includes a remote event control block that correlates to a event control block.

20. The node of claim 13, wherein said event server includes an event channel descriptor to access said event control block for said client application.
21. The node of claim 12, further comprising an event application program interface to publish and subscribe to said event channel.
22. The node of claim 12, wherein said event is processed by said application.
23. The node of claim 12, wherein said event is placed in a queue by said event server.
24. The node of claim 12, wherein said event includes at least one pattern field.
25. The node of claim 24, wherein said at least one pattern field matches at least one filter field within said filter.
26. The node of claim 12, wherein said filter is assigned by said application.
27. An event for exchanging information within a network, comprising:
at least one pattern field having information, said information to pertain to an application; and
a data field.

28. The event of claim 27, wherein said at least one pattern field matches a filter field assigned by said application.

29. The event of claim 27, wherein said event is placed on an event channel at a node within said network.

30. The event of claim 29, wherein an event server places said event on said event channel.

31. The event of claim 27, wherein said at least one pattern field is taken from a binary tree.

32. The event of claim 31, wherein said binary tree is a lexicographic binary tree.

33. A method for receiving information at a node, comprising:
opening an event channel at said node;
assigning a filter to said event channel; and
receiving an event on said event channel according to said filter.

34. The method of claim 33, further comprising registering a function indicating said opened event channel.

35. The method of claim 33, further comprising publishing said event on said event channel.

36. The method of claim 33, further comprising dispatching a callback responding to said event.

37. The method of claim 33, further comprising creating said event channel.

38. The method of claim 33, further comprising filtering said event by said filter.

39. The method of claim 38, wherein said filtering includes matching a pattern within said event with a filter pattern within said filter.

40. The method of claim 33, further comprising storing said event at an event control block.

41. The method of claim 33, further comprising using said event in a client application correlating to said event channel.

42. The method of claim 33, further comprising subscribing to said event on said event channel.

43. The method of claim 42, wherein said subscribing includes invoking an event control block.
44. A method for opening an event channel within a network, comprising:
receiving a request from an application on a node for said event channel;
registering a callback function from said application; and
subscribing to an event for said application on said event channel.
45. The method of claim 44, further comprising creating said event channel.
46. The method of claim 45, further comprising granting an authorization to said event channel.
47. The method of claim 45, further comprising registering said event channel.
48. The method of claim 44, further comprising publishing another event on said event channel.
49. A method for opening an event channel within a network, comprising:
receiving a request from an application on a node for said event channel;
registering a callback function from said application; and

publishing an event from said application on said event channel.

50. The method of claim 49, further comprising subscribing to another event on said event channel.

51. An event server on a node within a network, said network exchanging events over an event channel coupled to said node, comprising:

an event control block to receive events from said event channel;

5 an event channel descriptor to access said events from said event control block; and

an event control manager to update said event control block.

52. The event server of claim 51, wherein said event channel descriptor is allocated when said event channel is opened.

53. The event server of claim 52, wherein said event channel descriptor is de-allocated when said event channel is terminated.

54. The event server of claim 51, wherein said event control block tracks said event channel descriptor.

55. The event server of claim 51, wherein said event control block queues said events from said event channel.
56. The event server of claim 51, wherein said event channel descriptor accesses said event control block for an application.
57. The event server of claim 51, wherein said event control block includes a remote event control block object to connect with remote event control blocks within said network.
58. The event server of claim 57, wherein said event control manager updates a list having said remote event control block object.
59. The event server of claim 51, wherein said event control manager detects an overload condition in said event server.
60. The event server of claim 52, further comprising an event protocol module to collect said events.
61. The event server of claim 60, wherein said event protocol module filters said events.

62. A method for declaring a node to an event server, comprising:
granting access to an event channel;
5 updating an event control block in said event server reflecting said granted
access; and
sending a filter control message to another event server at another node.

63. The method of claim 62, further comprising allocating said event control
block.

64. The method of claim 62, further comprising finding said event control block
on said event server.

65. The method of claim 62, further comprising getting a naming context for said
event channel.

66. The method of claim 62, further comprising creating a naming context for
said event channel.

67. The method of claim 62, further comprising unlocking said event control
block.

68. The method of claim 62, further comprising changing an access permission to said event channel.

69. The method of claim 62, further comprising returning to an application at said node.

70. A method for implementing distributed filtering at a node, comprising:
building a filter from a plurality of search trees;
receiving an event at said node;
selecting a search tree from said filter; and
comparing said event with said search tree.

71. The method of claim 70, further comprising building said plurality of search trees.

72. The method of claim 71, further comprising placing heads from said plurality of search trees within said filter.

73. The method of claim 70, further comprising modifying said search trees.

74. A method for source filtering at an event server on a publisher node within a network, comprising:

sending a filter control message to said publisher node;

5 marking a remote event control block object in an event control block

according to said filter control message; and

filtering events from said event control block.

75. The method of claim 74, further comprising building said filter control message.

76. The method of claim 75, further comprising selecting a filter search tree.

77. The method of claim 76, further comprising modifying said filter search tree.

78. The method of claim 74, further comprising changing access permissions to a remote event control block and re-sending said filter control message.

79. The method of claim 74, further comprising unmarking said remote event control block object.

80. A method for receiving information at a node, comprising:

receiving an event from an event channel at said node; and

filtering said event from said event channel with a filter corresponding to an application at said node.

81. The method of claim 80, further comprising assigning said filter to said event channel by said client application.

82. The method of claim 80, further comprising opening said event channel by said client application.

83. The method of claim 82, further comprising opening said event channel in a write mode.

84. The method of claim 82, further comprising opening said event channel in a read mode.

85. The method of claim 80, further comprising queuing said event in an event control block corresponding to said application.

86. The method of claim 80, further comprising dropping said event during an overload condition at said node.

87. A computer program product comprising a computer useable medium having computer readable code embodied therein for receiving information at a node, the computer program product adapted when run on a computer to effect steps

5 including:

- opening an event channel at said node;
- assigning a filter to said event channel; and
- receiving an event on said event channel according to said filter.

88. A computer program product comprising a computer useable medium having computer readable code embodied therein for opening an event channel within a network, the computer program product adapted when run on a computer to effect steps, including:

- 5
- receiving a request from an application on a node for said event channel;
 - registering a callback function from said application; and
 - subscribing to an event for said application on said event channel.

89. A computer program product comprising a computer useable medium having computer readable code embodied therein for opening an event channel within a network, the computer program product adapted when run on a computer to effect steps including:

- 5
- receiving a request from an application on a node for said event channel;
 - registering a callback function from said application; and

publishing an event from said application on said event channel.

90. A computer program product comprising a computer useable medium having computer readable code embodied therein for declaring a node to an event server, the computer program product adapted when run on a computer to effect steps including:

- 5 granting access to an event channel;
- updating an event control block in said event server reflecting said granted access; and
- sending a filter control message to another event server at another node.

91. A computer program product comprising a computer useable medium having computer readable code embodied therein for implementing distributed filtering at a node, the computer program product adapted when run on a computer to effect steps, including:

- 5 building a filter from a plurality of search trees;
- receiving an event at said node;
- selecting a search tree from said filter; and
- comparing said event with said search tree.

92. A computer program product comprising a computer useable medium having computer readable code embodied therein for source filtering at an event server on a

publisher node within a network, the computer program product adapted when run

5 on a computer to effect steps, including:

sending a filter control message to said publisher node;

marking a remote event control block object in an event control block

according to said filter control message; and

filtering events from said event control block.

93. A computer program product comprising a computer useable medium having
computer readable code embodied therein for receiving information at a node, the
computer program product adapted when run on a computer to effect steps
including:

receiving an event from an event channel at said node; and

filtering said event from said event channel with a filter corresponding to an
application at said node.